REMARKS

Claims 1, 19 and 20 are amended. Claims 12 and 13 are canceled. Claims 14-18 and 21-22 are withdrawn from further consideration. Therefore, claims 1-11 and 19-20 are pending in this application.

Claim Rejections Under 35 USC §102(e)

Claims 1-13 and 19-20 were rejected under 35 U.S.C. 102(e) as being anticipated by *Chu et al.* US Patent No. 6,718,376 ("Chu"). In response, claims 1, 19 and 20 are amended. Claim 20 is further amended to correct a typographical error. Claims 12 and 13 are canceled.

The Applicants believe that amended claims 1, 19 and 20 are patentable over and not anticipated by Chu. Due to the complexity and broadness of dependencies between service component entries in the pending application, a more sophisticated management technique is required and disclosed. Chu limits his/her dependency configuration and does not disclose the same technique. This distinction is detailed in the following remarks.

Chu states in col. 7, lines 34-37 and lines 45-48 that a control adapter is always responsible for starting and stopping a service on the same node. Each service thus depends on only the control adapter, a simple one-tiered dependency.

In the pending application, however, the dependencies are not limited to only one control adapter creating services. They may be multi-tier and more complex. For instance, in Figure 4A, both *services* and *svchost 1* must be created and running in order for *svchost2* to start. *svchost2* - and therefore, by dependency, both *services* and *svchost1* - must be created and running in order for *lsass* to start. Other complex dependency examples are illustrated in Figure 4a and in the table format of Figs. 3A and 3B. The nature of dependency management

may be multi-tiered and in some cases, cross-tiered. It is much more complex in the pending application than the simple one-tiered dependency, as disclosed by Chu.

The allowance for complex dependencies necessitates a more sophisticated management technique which the pending application provides over Chu. Chu's technique for discovering non-operational network services consists of having the control adapter poll each service on a prescribed time basis (Chu, col. 6, lines 10-13, col. 9, lines 38-42, col. 10, lines 29-31). Chu, col. 9, lines 38-40 describes the process as "the control adapter ... constantly polling the service adapter to insure that the service adapter is functional." The control adapter dialogues constantly with each service adapter to ensure live operation of that service adapter. Chu's technique suffices for the simple one-tiered dependency environment.

Using Chu's technique in the complex dependency environment of the pending application, however, would be unwieldy and produce a large volume of message traffic from the constant polling of the multiple tiers and dependent services. Claims 1, 19 and 20 are amended to distinctly point out a more efficient technique over Chu as required by the environment of the pending application. First, maintaining a dynamic service consistency file and a reference file (i.e., a representation of the current and desired complex dependencies) is disclosed by amended claims 1, 19 and 20. By adding the limitations of canceled claims 12 and 13 - "comprising a ... representation of a tree, a graph, a linked list, or a table" (specification page 9, lines 21-14), multi-tiered, complex dependencies may be represented efficiently. The knowledge of multi-tiered dependencies is important for determining and rectifying the chain of events in an error condition, for instance, to accurately attribute a lost process to the actual failed process in a multi-tiered dependency chain. These multi-tiered

mapped dependency representations are not discussed in Chu, as Chu limits his/her dependencies to a single tier relationship (Chu, col. 7, lines 34-37 and lines 45-48).

Second, a more efficient and applicable technique for discovering non-operational operating system services is disclosed by amended claims 1, 19 and 20. This technique or method compares a dynamic representation of multi-tier, inter-dependent services against a reference representation of said services, i.e., the dynamic service consistency file against the reference file. By using a single file comparison, all currently non-operational operating system services in the multi-tiered, complex dependency environment may be determined. Chu does not disclose a single file comparison of complex mapped dependency representations to find non-operational services. Chu requires a control adapter to sequentially poll each network service adapter - a longer procedure requiring more time, more messaging, and the absence of a roadmap extendable to represent a complex dependency web of services.

Summary:

Amended claims 1, 19 and 20 disclose representations for mapping complex service dependencies and a file comparison method for determining the services that are not operational. Chu does not provide this disclosure – (s)he discloses an environment with only one tier of dependency and a polling mechanism to determine non-operational network services. Furthermore, the pending application is patentable over Chu as it discloses a unique way to represent complex dependencies between services and a more efficient method for determining inconsistencies in a complex dependency environment by using a single file

comparison. Accordingly, amended claims 1, 19 and 20 are not anticipated by Chu and have benefit and patentability over Chu.

As claims 2-11 depend on amended independent claim 1 and claim 1 is not anticipated by Chu, claims 2-11 are also allowable.

CONCLUSION

In view of the above amendment and arguments, the applicant submits the pending application is in condition for allowance and an early action so indicating is respectfully requested.

The Commissioner is authorized to charge any fee deficiency required by this paper, or credit any overpayment, to Deposit Account No. 13-2855, under Order No. 30835/303495, from which the undersigned is authorized to draw.

Dated: October 11, 2007

Respectfully submitted,

By___/W. J. Kramer #46,229/_____
William J. Kramer
Registration No.: 46,229
MARSHALL, GERSTEIN & BORUN LLP
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Attorney for Applicant